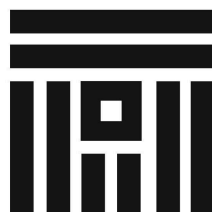


UNIVERSITATEA TRANSILVANIA DIN BRAȘOV



FACULTATEA DE INGINERIE ELECTRICĂ ȘI ȘTIINȚA CALCULATOARELOR

DEPARTAMENT AUTOMATICĂ ȘI INFORMATICĂ APLICATĂ

Proiect *Sisteme inteligente de control*

Tema nr. 59

Autor

Andrei VASILCOI

Mai 2019, Brașov

Cuprins

1	Tema proiectului	2
2	Indicații și recomandări	2
3	Conținutul minimal al proiectului	2
4	Condiții pentru susținerea proiectului	3
5	Considerații teoretice	4
5.1	Introducere în logica Fuzzy	4
5.2	Interferente de tip Sugeno-Takagi	4
5.3	Structura sistemului de reglare	5
6	Rezolvare	5
7	Concluzii	6

1 Tema proiectului

Se va proiecta un regulator fuzzy pentru controlul unui proces descris de funcția de transfer:

$$G_p(s) = \frac{K_p}{(sT_1 + 1)(sT_2 + 1)} \quad (1)$$

Se va realiza o implementare a regulatorului proiectat, în limbajul C/C++, care ulterior se va testa pentru diverse valori ale mărimilor de intrare ale sistemului de inferență prin comparație cu funcționarea aceluiași sistem de inferențe realizat în Matlab.

Tab. 1: Date de proiectare

Tema nr.	Parametrii procesului			M_v	t_s	e_{st}	Student
	K_p	T_1	T_2				
59	1	120	5	4%	120	0%	Vasilcoi S. Andrei

2 Indicații și recomandări

- Proiectarea și analiza sistemului de reglare se va face folosind mediul Matlab-Simulink. Implementarea în program se va face doar după determinarea sistemului de inferențe cu care se obțin performanțele impuse.
- Nu sunt impuse restricții pentru tipul regulatorului fuzzy sau tipul inferențelor utilizate, decât aceea că alegerea acestora trebuie să conducă la obținerea performanțelor.
- Programul realizat va fi testat prin comparație cu funcționarea sistemului implementat în mediul Matlab. Se vor lua diverse valori ale variabilelor de intrare ale sistemului de inferență pentru care se vor înregistra valorile de ieșire obținute în Matlab și cele obținute cu ajutorul programului.

3 Conținutul minimal al proiectului

- Enunțul și descrierea problemei.
- Prezentarea sistemului de reglare și a structurii regulatorului fuzzy ales. Obs: Se va justifica alegerea tipului de regulator.
- Prezentarea sistemului de inferență obținut (variabile fuzzy, funcții de apartenență, reguli, tipul inferenței fuzzy, caracteristica statică, metoda de defuzzificare – dacă este cazul – etc.). Obs: Se vor justifica toate alegerile referitoare la sistemul de inferență.
- Prezentarea rezultatelor simulării sistemului de reglare proiectat (Matlab-Simulink).
- Prezentarea particularităților de implementare, utilizate la realizarea programului.
- Se vor anexa schema Simulink și programul realizat.

4 Condiții pentru susținerea proiectului

- a) Proiectul va fi prezentat și susținut. Se vor adresa întrebări referitoare la noțiuni de bază și aspecte întâlnite în realizarea lui.
- b) Proiectul predat trebuie să conțină: redactarea (format electronic și printat), fișierele obținute cu ajutorul aplicației de editare a sistemelor de inferență fuzzy, schemele Simulink, codul sursă al programului și fișierul executabil.
- c) Proiectul poate fi predat până la data examenului scris.

5 Considerații teoretice

5.1 Introducere în logica Fuzzy

Abordarea inginerască „clasică”, strictă, a realității este una în special cantitativă, bazată pe modelări matematice exprimate în forme care inspiră exactitatea. Într-o astfel de abordare, aprecierea de ordin calitativ a mărimilor, valorilor, rezultatelor etc. este greu interpretabilă, deoarece acestea au valori stricte, bine precizate. Într-un context strict, modelele disponibile sunt exacte (de fapt cât mai exacte posibil, cu aproximări și supoziții), iar sistemele de comandă și control sunt dezvoltate în strânsă legătură cu acestea.

Aplicațiile practice în domeniul ingineriei electrice, bazate pe teoria mulțimilor fuzzy, între care și controlul fuzzy, includ un mecanism de evaluare numerică a aprecierilor calitative, proprii exprimării uzuale ale experienței. Abordarea problemelor se extinde, incluzând și aprecieri calitative ale rezultatelor.

Aplicațiile curente în inginerie și mai exact în control automat apelează doar o parte restrânsă a teoriei mulțimilor fuzzy, care se dovedește relativ ușor accesibilă.

O propoziție fuzzy simplă este o sintagmă de forma “este”, în care x este o variabilă fuzzy, iar A este un termen lingvistic (atribut) ce poate caracteriza variabila. Prin analogie, o propoziție strictă este descrisă printr-o egalitate. Propozițiile fuzzy conțin cuvântul “este”, care ține loc semnului “=” din relațiile stricte.

Valoarea de adevăr a propoziției fuzzy este descrisă de funcția de apartenență a mulțimii fuzzy A , adică, pentru o valoare strictă putem preciza în ce măsură “este” prin valoarea lui.

5.2 Interferente de tip Sugeno-Takagi

În unele aplicații practice concrete, procedura stabilirii unei baze de reguli poate consta într-o procesare a unui set de date non-fuzzy (reale). Finalitatea acestei prelucrări ar putea fi transpunerea acestor date într-o formă lingvistică. Datele non-fuzzy sunt obținute prin înregistrarea semnalelor utile pentru proiectare - mărime de comandă, mărime de reacție și/sau eroare de reglare – și prin calcularea derivatelor acestora, în funcție de structura de reglare aleasă. Însă extragerea regulilor fuzzy în forma “dacă ... atunci ...” este considerată dificilă din cauză faptului că nu există proceduri sistematizate.

Obținerea regulilor este strâns legată de mecanismul de inferență utilizat. Dacă un set de date reale privind funcționarea procesului este disponibil, se poate adapta mecanismul de inferență fuzzy de tip Tagaki, într-o formă utilă și adecvată datelor numerice. Această adaptare reușește să reducă din volumul de calcule al metodelor de inferență obișnuite, prin introducerea unei reprezentări mai degrabă numerice a regulilor fuzzy, care în final nu mai sunt deloc evidențiate.

Forma generală a regulilor fuzzy de tip Sugeno este:

$$\text{daca } x_1 \text{ este } A_{11} \text{ si } x_2 \text{ este } A_{21} \text{ este atunci } y_1 = f_1(x_1, x_2), \quad (2)$$

unde y_1 este o valoare crisp (singleton, reală). De obicei $f_1(x_1, x_2)$ este un polinom cu variabilele x_1 și x_2 de forma $y_1 = p_1x_1 + q_1x_2 + r_1$. p_1 , q_1 și r_1 sunt constante (date, cu valori cunoscute).

Indicele inferior 1 se referă la numărul regulii într-o bază de reguli.

O regulă fuzzy Sugeno poate fi:

- de ordinul zero (daca $p_i, q_i = 0$)
- de ordinul unu (daca p_i sau $q_i \neq 0$)

- de ordinul doi (daca p_i sau $q_i \neq 0$)

5.3 Structura sistemului de reglare

Se consideră structura clasică de reglare, cu un singur regulator principal, aflat pe calea directă a sistemului. Această structură permite o comparație ușoară a performanțelor reglatoarelor fuzzy cu cele ale reglatoarelor clasice, evident pe baza cunoștințelor prealabile necesare de ingineria reglării automate.

Structura de reglare convențională este un sistem cu reacție negativă, considerată unitară pentru o simplificare inițială. Pe calea de reacție este preluată mărimea de ieșire a părții fixe $y(t)$, care practic este un semnal achiziționat corespunzător mărimii supuse reglării. Mărimea de ieșire măsurată este comparată cu o mărime de referință $r(t)$, care definește evoluția dorită în timp a sistemului, iar diferența celor două reprezintă eroarea de reglare:

$$e(t) = r(t) - y(t) \quad (3)$$

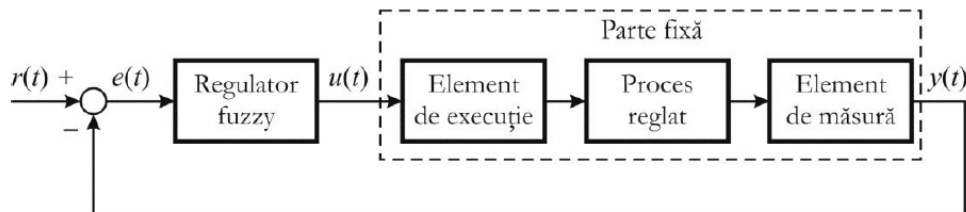


Fig. 1: Structura conventionala de reglare

6 Rezolvare

7 Concluzii

Din punct de vedere al ușurinței de proiectare criteriul modului varianta Kessler este cel mai accesibil algoritm. Însă performanțele obținute cu acest regulator nu au îndeplinit cerința timpului de stabilire de 3 secunde sau cea a suprareglajului de 4.56%. Din acest motiv s-a pornit de la regulatorul PI obținut prin varianta Kessler, s-a adăugat o componentă derivativă pentru a micșora suprareglajul, s-a modificat de asemenea și T_i pentru a reduce timpul de stabilire și s-a ajuns la o performanță foarte apropiată de cea impusă, timpul de stabilire fiind mai lung cu 0.2 secunde.

Datorită îndeplinirii criteriilor de performanță, regulatorul obținut prin încercări are și implementare în limbajul C, deși algoritmul propus poate fi aplicat foarte ușor pentru orice ecuație cu diferențe, fiind generic.

Pentru metoda experimentală Ziegler-Nichols s-au obținut cele mai slabe performanțe iar metoda de proiectare este relativ dificilă dacă nu avem instrumentele adecvate pentru a determina panta și pentru a găsi punctul de inflexiune din datele achiziționate. Odată obținute însă valorile α și L se pot încerca diferite tipuri de regulatoare având de făcut niște substituții simple.

Regulatorul proiectat după stare a fost cel mai ușor de implementat și testat cu programul Matlab. Singurul impediment este constituit de alegerea polilor conjugați și găsirea unei metode algoritmice de a-i găsi. Metoda aplicată în acest proiect se folosește de forma generală a unui sistem de gradul II și indicatorii de calitate la mărimea de intrare treaptă unitară.

Bibliografie

- [1] C. Boldișor, *Ingineria Reglării Automate - note de curs.*
- [2] S. Coman, *Teoria Sistemelor I și II - note de curs.*
- [3] S. Coman, *Sisteme Automate cu Eșantionare - note de curs.*